Binomial random variables

---

56 marks

Suppose $X \sim Binomial(n, p)$, then

$$Pr(X = x) = \left(\begin{array}{c} n \\ x \end{array} \right) p^x (1-p)^{n-x} \qquad\qquad x = 0, 1, \ldots, n.$$

from which it follows that $E(X) = np$ and $Var(X) = np(1-p)$.

Once x is observed, the unknown proportion can be estimated (e.g. via maximum likelihood) as the numerical value of $\widehat{p} = x/n$; the corresponding random estimat**or** would be $\widetilde{p} = X/n$.

In this question, you are going to develop your intuition about simple proportions as random variables through a bit of mathematics, some introductory R programming, and a little simulation and visualization.

a. The **odds** (in favour of the event) are defined by the ratio $p/(1-p)$ (e.g. even odds are 1, or 1:1, when $p = 1/2$; odds are 9:1 in favour when $p = 9/10$; odds less than one are often inverted to be described as the odds *against* the event).

   Suppose we are interested in comparing the binomial probabilities $Pr(X = a)$ to $Pr(X = b)$. For example, the ratio, $Pr(X = a)/Pr(X = b)$, tells us how many times more (or less) likely it is to observe $x = a$ than $x = b$.

   i. *(1 mark)* Express the ratio $Pr(X = a)/Pr(X = b)$ as a function of the odds.

$$\frac{Pr(X=a)}{Pr(X=b)} = \frac{\binom{n}{a} p^a (1-p)^{n-a}}{\binom{n}{b} p^b (1-p)^{n-b}} = \frac{\binom{n}{a}}{\binom{n}{n}} \frac{p^{a-b}}{(1-p)^{-n+a+n-b}} = \frac{\binom{n}{a}}{\binom{n}{b}} \frac{p^{a-b}}{(1-p)^{a-b}} = \frac{\binom{n}{a}}{\binom{n}{b}} odd^{a-b}$$

   ii. *(4 marks)* Here, in two different ways, you will write a function of the odds which calculates $Pr(X = a)/Pr(X = b)$ for a given n.

   For the first way, write `prob_ratio1()` to do the calculation using the function `choose()`:

```
prob_ratio1 <- function (n, a, b, odds = 1) {
   choose_n_a <- choose(n,a)
   choose_n_b <- choose(n,b)
   ans <- choose_n_a/choose_n_b*odds^(a-b)
   return(ans)
}
# print(prob_ratio1(5,3,4,5))
```

```
For the second way, write `prob_ratio1()` to do the calculation using the function `dbinom()`:
```

```
prob_ratio2 <- function (n, a, b, odds = 1) {
  p <-  odds/(1+odds)
  p_a <- dbinom(a,size=n,prob=p)
  p_b <- dbinom(b,size=n,prob=p)
  return(p_a/p_b)
}
# print(prob_ratio2(5,3,4,5))
```

```
Both are calculating the same values.
```

```
iii. *(2 marks)* Report the following values for `prob_ratio1()` and `prob_ratio2()`
```

```r
# using choose()
prob_ratio1(50, a = 5, b = 45)
```

```
## [1] 1
```

```r
prob_ratio1(50, a = 5, b = 45, odds = 9)
```

```
## [1] 6.765496e-39
```

```r
# and using dbinom()
prob_ratio2(50, a = 5, b = 45)
```

```
## [1] 1
```

```r
prob_ratio2(50, a = 5, b = 45, odds = 9)
```

```
## [1] 6.765496e-39
```

b. Extreme proportions like $\hat{p} \approx 0$ or $\hat{p} \approx 1$ often generate a great deal of interest in an analysis.

   For example, p might be the proportion of people in some population who perhaps die from some exposure to some toxin, or are cured of a disease by some treatment, or maybe just say they would vote for a particular party or candidate. In any of these cases it can be surprising (even alarming) to see either $\hat{p} \approx 0$ or $\hat{p} \approx 1$, so much so that some explanation seems in order.

   Suppose we have observed x from $Binomial(n, p)$ and y from $Binomial(m, p)$ – that is the same probability of occurrence but different sample sizes. Denote the observed proportions as $\hat{p}_x = x/n$ and $\hat{p}_y = y/m$, respectively.

   i. *(2 marks)* Give the mathematical expression for the ratio

   $$\frac{Pr(\tilde{p}_x = 0)}{Pr(\tilde{p}_y = 0)}$$

   and for the ratio

   $$\frac{Pr(\tilde{p}_x = 1)}{Pr(\tilde{p}_y = 1)}$$

   .

   $$\frac{Pr(\tilde{p}_x = 0)}{Pr(\tilde{p}_y = 0)} = \frac{\binom{n}{0} p^0 (1-p)^n}{\binom{m}{0} p^0 (1-p)^m} = \frac{\binom{n}{0}}{\binom{m}{0}} (1-p)^{n-m} = (1-p)^{n-m}$$

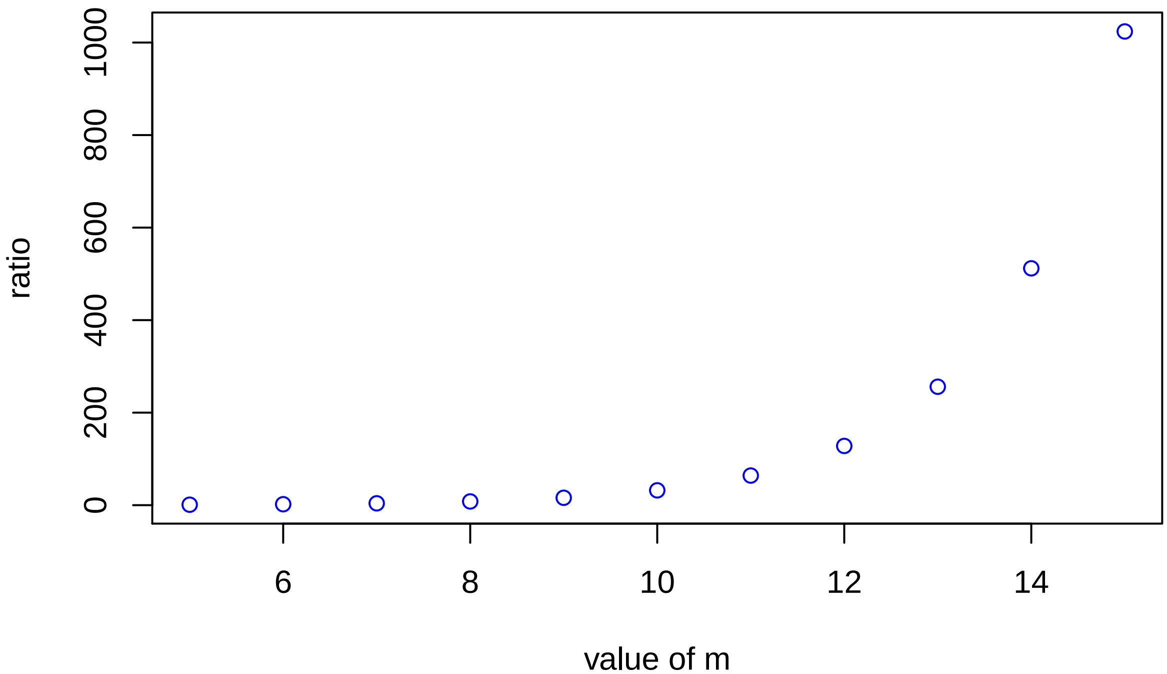
   $$\frac{Pr(\tilde{p}_x = 1)}{Pr(\tilde{p}_y = 1)} = \frac{\binom{n}{n} p^n (1-p)^0}{\binom{m}{m} p^m (1-p)^0} = \frac{\binom{n}{n}}{\binom{m}{m}} p^{n-m} = p^{n-m}$$

   ii. *(2 marks)* Describe what happens to each of these ratios for $n < m$ and m increases.

   - As m increases, the first ratio will increase. The base 1-p is between 0 and 1 and the exponent decreases from 0. The power will increase if m increases.
   - As m increases, the second ratio will increase. The base p is between 0 and 1 and the exponent decreases from 0. The power will increase if m increases.

   iii. *(3 marks)* Suppose $p = 1/2$, $n = 5$ and $m \in \{5, 6, 7, \ldots, 14, 15\}$. Using the `plot()` function (with appropriate title and axis labels), plot the curve of the pair (n, m) for all values.

```
ms <- seq(from=5, to=15)
get_ratio <- function(p,n,ms){
    ans = c()
    for (i in 1:length(ms)){
        ans <- c(ans, p^(n-ms[i]))
    }
    return(ans)
}
ratios <- get_ratio(1/2,5,ms)
plot(ms,ratios,xlab="value of m",ylab="ratio",col="blue")
```




```
iv. *(2 marks)* In plain English, express when $\widehat{p}$ is most likely to be 1 and when you should
```

c. For the binomial definition, as given above:

i. *(2 marks)* Mathematically derive $E(\tilde{p})$ and the **standard deviation** $SD(\tilde{p})$.

$$\tilde{p} = \frac{X}{n}$$

$$E\left(\tilde{p}\right) = E\left(\frac{1}{n} \cdot X\right) = \frac{1}{n}E\left(X\right)$$

$$E\left(X\right) = n \cdot p$$

$$E\left(\tilde{p}\right) = \frac{1}{n}E\left(n \cdot p\right) = \frac{1}{n} \cdot n \cdot E\left(p\right) = p$$

$$Var(\tilde{p}) = Var\left(\frac{X}{n}\right) = \frac{1}{n^2}Var(X)$$

$$Var(X) = E\left(X^2\right) - E^2\left(X\right) = np(1-p)$$

$$Var\left(\tilde{p}^2\right) = Var\left(\frac{X}{n}\right) = \left(\frac{1}{n}\right)^2 Var\left(X\right) = \frac{1}{n^2}np(1-p) = \frac{p(1-p)}{n}$$

$$SD\left(\tilde{p}\right) = \sqrt{Var\left(\tilde{p}\right)} = \sqrt{\frac{p(1-p)}{n}}$$

ii. *(2 marks)* Write a function that calculates the standard deviation of $\tilde{p}$ for any pair of values of n and p as follows:

```r
sd_p_wig <- function (n, p) {
    # Your code here
    return(sqrt(p*(1-p)/n))
}
```

d. Chebyshev's inequality relates the nearness of a random variable Y to its expectation μ as a function of its standard deviation σ (provided both exist and are finite) as follows:

$$Pr\left(|Y-\mu| \geq k\sigma\right) \leq \frac{1}{k^2}$$

for any constant $k > 1$.

i. *(2 marks)* Write this inequality when $Y = \tilde{p}$, the binomial proportion estimator.

$$\mu = E\left(\tilde{p}\right) = \frac{X}{n} = p \qquad Y = \tilde{p} \qquad \sigma = SD(\tilde{p}) = \sqrt{\frac{p(1-p)}{n}}$$

$$Pr\left(|Y-\mu| \geq k\sigma\right) \leq \frac{1}{k^2} = Pr\left(|\tilde{p}-p| \geq k\sqrt{\frac{p(1-p)}{n}}\right) \leq \frac{1}{k^2}$$

ii. *(4 marks)* Suppose that $k = 5$. As a function of p, mathematically express the sample size n needed to ensure by Chebyshev's inequality that for our estimator $\tilde{p}$ we have

$$Pr\left(|\tilde{p}-p| \geq \frac{1}{50}\right) \leq \frac{1}{25}.$$

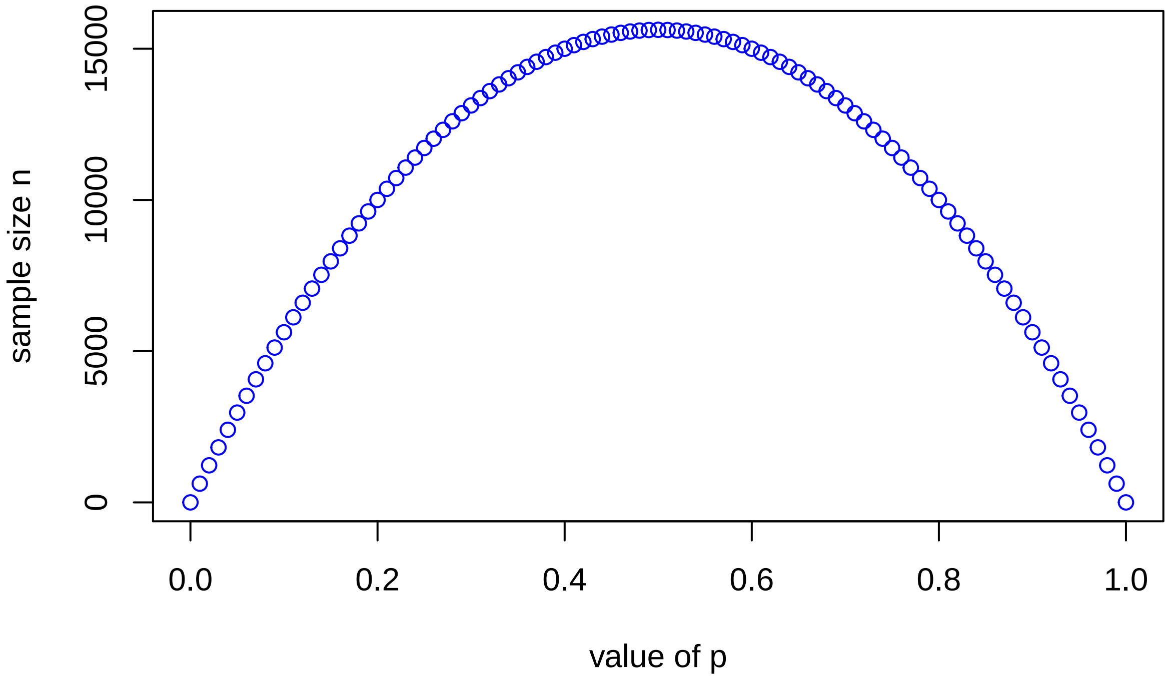
$$k\sqrt{\frac{p(1-p)}{n}} = 1/50$$

$$k = 5$$

$$\sqrt{\frac{p(1-p)}{n}} = \frac{1}{250}n = 250^2 p\left(1-p\right)$$

Using the function `plot()`, with appropriate title and x and y axis labels, plot the curve of n as a function of p for `p = seq(0,1, 0.01)`.

```r
p <- seq(0,1,0.01)
get_n <- function(p_list){
    ans = c()
    for (i in 1:length(p_list)){
        ans <- c(ans, ((250^2)*p[i]*(1-p[i])))
    }
    return(ans)
}
n_list <- get_n(p)
plot(p,n_list,xlab="value of p",ylab="sample size n",col="blue")
```



iii. *(4 marks)* Suppose that $k = 5$, and $n = 2500$. As a function of p, mathematically express the bound B given by Chebyshev's inequality, so that for our estimator $\tilde{p}$ we have

$$Pr\left(|\tilde{p} - p| \geq B\right) \leq \frac{1}{25}.$$

Using the function `plot()`, with appropriate title and x and y axis labels, plot the curve of B as a function of p for `p = seq(0,1, 0.01)`. iv *(2 marks)* In simple English, summarize what the largest B says about how well p is likely to be estimated when $n = 2500$ according to Chebyshev's inequality.

$$Pr(|\tilde{p} - p| \geq B) \leq \frac{1}{25}$$

$$Pr(|\tilde{p} - p| \geq k\sqrt{\frac{p(1-p)}{n}}) \leq \frac{1}{25}$$

$$B = k\sqrt{\frac{p(1-p)}{n}} \qquad k = 5 \qquad n = 2500$$

$$B = \frac{\sqrt{p(1-p)}}{10}$$

d. The functions `plot()`, `lines()`, and `abline()` can be used to plot some data, to add curves, and to add a straight line to a plot. To learn more, type `help("plot.default")`, `help("lines")`, and `help("abline")`.

`Use these plotting functions, and appropriate arguments (including meaningful titles, and axis labels),`

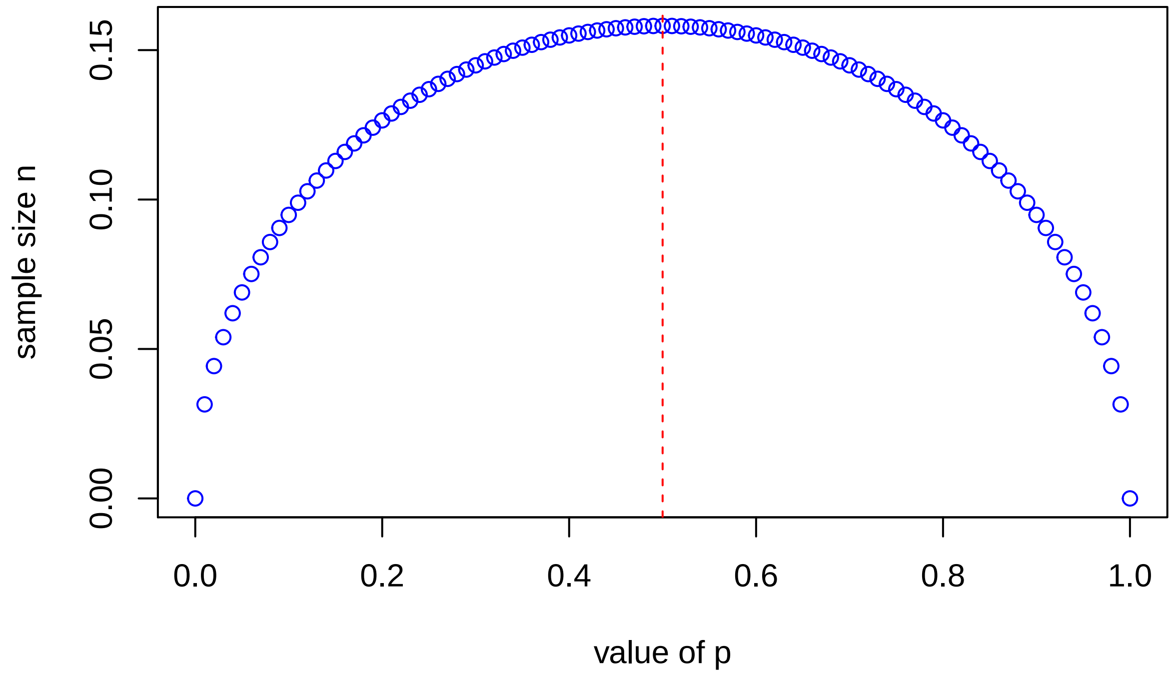
i. *(3 marks)* As a function of `p = seq(0, 1, 0.01)` for a fixed `n = 10`. Add a vertical **dashed** line in **red** at the value of p which maximizes the standard deviation.

`Hand in your code and plot.`

```
p_list <- seq(0, 1, 0.01)
n <- 10
get_sd <- function(n,list_p){
  ans = c()
  for (i in 1:length(p_list)){
```

```r
      ans <- c(ans, sqrt(p[i]*(1-p[i])/n))
    }
  return(ans)
}
sd_list <- get_sd(n,p_list)
max_index = which.max(sd_list)

plot(p_list,sd_list,xlab="value of p",ylab="sample size n",col="blue")
abline(v=p_list[max_index],lty=2,col="red")
```



ii. *(4 marks)* As a function of n $\in \{5, 10, 15, \ldots, 50\}$ for the different values of p $\in \{0.1, 0.3, 0.5, 0.8\}$.

```
- Use a different colour and line type for each curve (i.e. value of `p`).
- Use `lwd = 2` for all curves.
- Use `legend()` to add a legend to the `"topright" corner of the plot,  identify each curve by its
- Hand in your code and plot.
```

```r
n_list = seq(5,50,5)
p_list = c(0.1,0.3,0.5,0.8)
colors = c("red","green","pink","maroon")

get_sd <- function(n_list,p_list){
  ans_list = c()
  for(i in 1:length(p_list)){
    ans <- c(sapply(n_list, function(n) sqrt(p_list[i]*(1-p_list[i])/n)))
      ans_list <- c(ans_list,ans)
      print(typeof(ans))
  }
  return(ans_list)
}


vals<-get_sd(n_list, p_list)
```

```
## [1] "double"
## [1] "double"
## [1] "double"
## [1] "double"
```

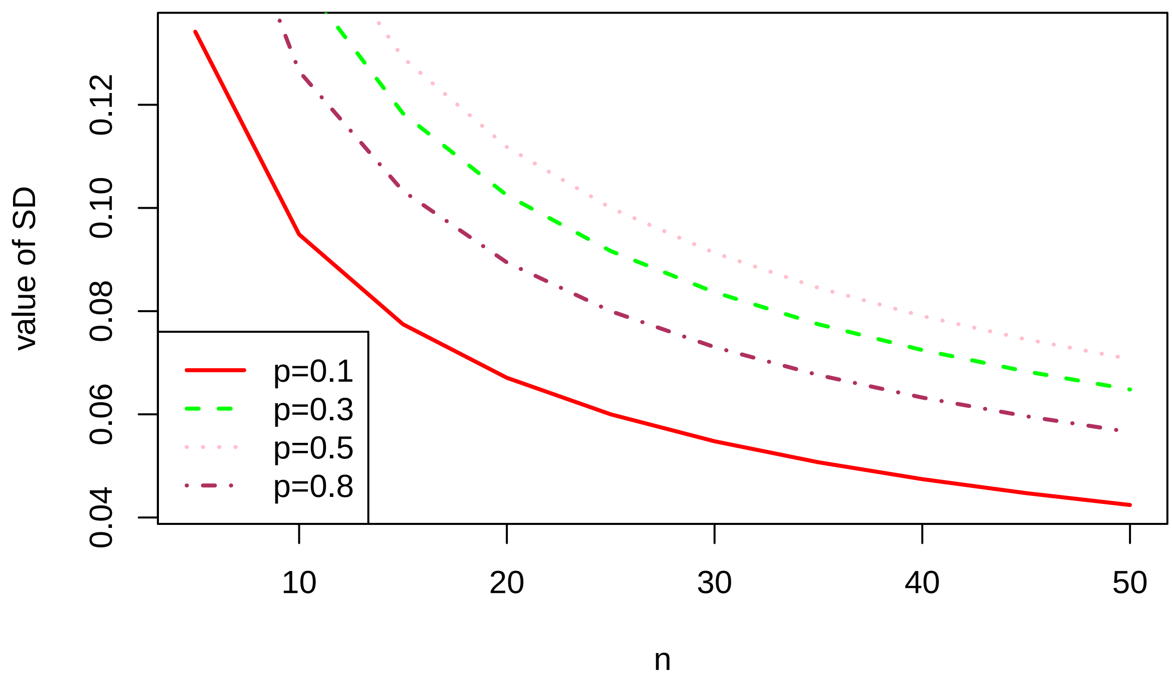
```r
dim(vals)<-c(10,4)

plot(n_list,vals[,1],lwd = 2,lty=1, type="l",col=colors[1],xlab="n",ylab="value of SD")
for(i in 2:length(p_list)){
  print(length(n_list))
  print(vals[,i])
  lines(n_list,vals[,i],lwd = 2,lty=i, type ="l",col=colors[i])

}
```

```
## [1] 10
##  [1] 0.20493902 0.14491377 0.11832160 0.10246951 0.09165151 0.08366600
##  [7] 0.07745967 0.07245688 0.06831301 0.06480741
## [1] 10
##  [1] 0.22360680 0.15811388 0.12909944 0.11180340 0.10000000 0.09128709
##  [7] 0.08451543 0.07905694 0.07453560 0.07071068
## [1] 10
##  [1] 0.17888544 0.12649111 0.10327956 0.08944272 0.08000000 0.07302967
##  [7] 0.06761234 0.06324555 0.05962848 0.05656854
```

```r
legend("bottomleft",
  legend = lapply(p_list,  function(x) paste("p=",x,sep="")),
  #legend = p_list,
  col = colors,
  lty=c(1,2,3,4),
  lwd=2
)
```



iii. *(2 marks)* Comment on your findings about the dependency of the standard deviation of the binomial proportion estimator $\tilde{p}$ as a function of n and p.

e. The R function `rbinom()` can be used to generate pseudo-random values `x` from a binomial distribution. In this question, you will examine the samples of the sample **proportions**, x/n, drawn from binomials with the same value of p but different values of n.

   i. *(3 marks)* Using `rbinom()`, for every $n \in \{5, 10, 15, \ldots, 45, 50\}$ generate 100 proportions, each one based on an independent pseudo-random value from $Binomial(n, 0.5)$.

      You might find the `rep()` function useful in constructing `n`.

      Show your code.

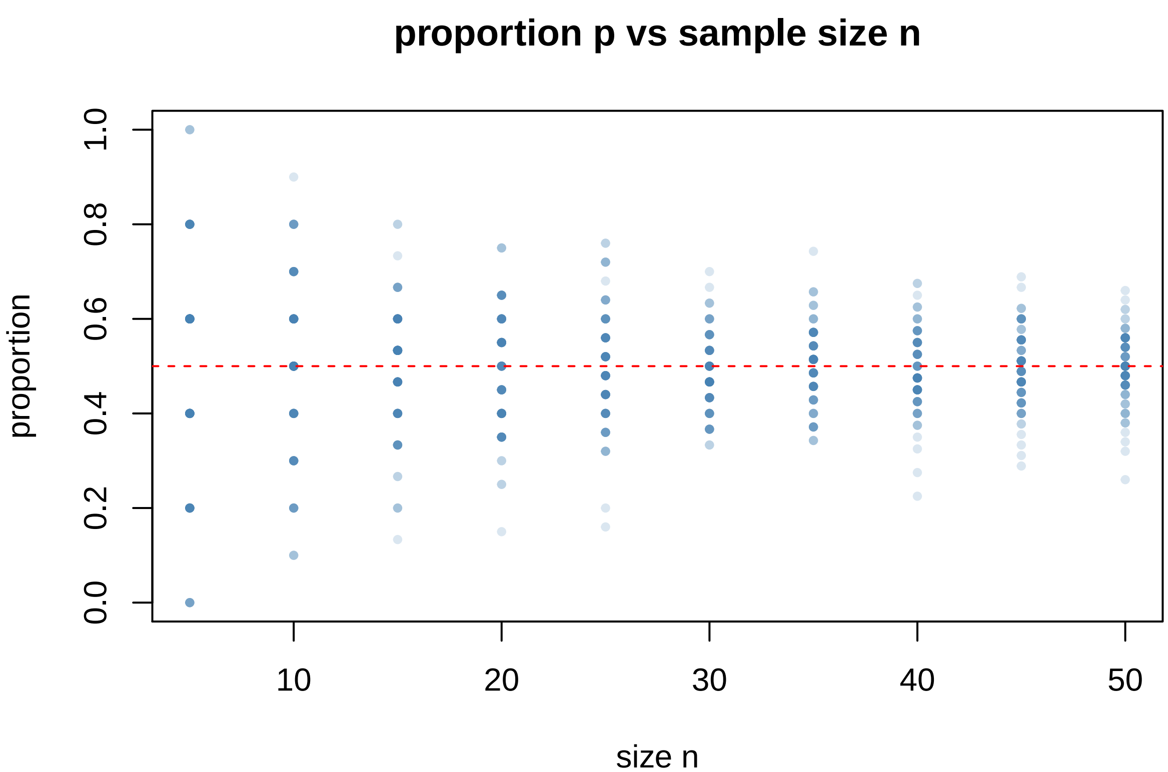
```
n_list = seq(5,50,5)
vals<- sapply(n_list, function(n) rbinom(100,n,0.5)/n)
```

```
ii. *(3 marks)* Plot the pairs $(n, \widehat{p})$ as points (use plot arguments `ylim = c(0,1), pch = 19

    Note that there will be 100 proportions for every $n$.

    Add a red dashed horizontal line at $p = 0.5$.
```

```
n_list = seq(5,50,5)
vals<- sapply(n_list, function(n) rbinom(100,n,0.5)/n)
vals_for_plot = unlist(vals)
n_for_plot = rep(seq(5,50,5),each = 100)
plot(n_for_plot, vals_for_plot, ylim = c(0,1), pch = 19, col =adjustcolor("steelblue", 0.2), cex = 0.5,
     xlab = "size n",ylab = "proportion",main="proportion p vs sample size n")
abline(h=0.5,lty=2,col="red")
```



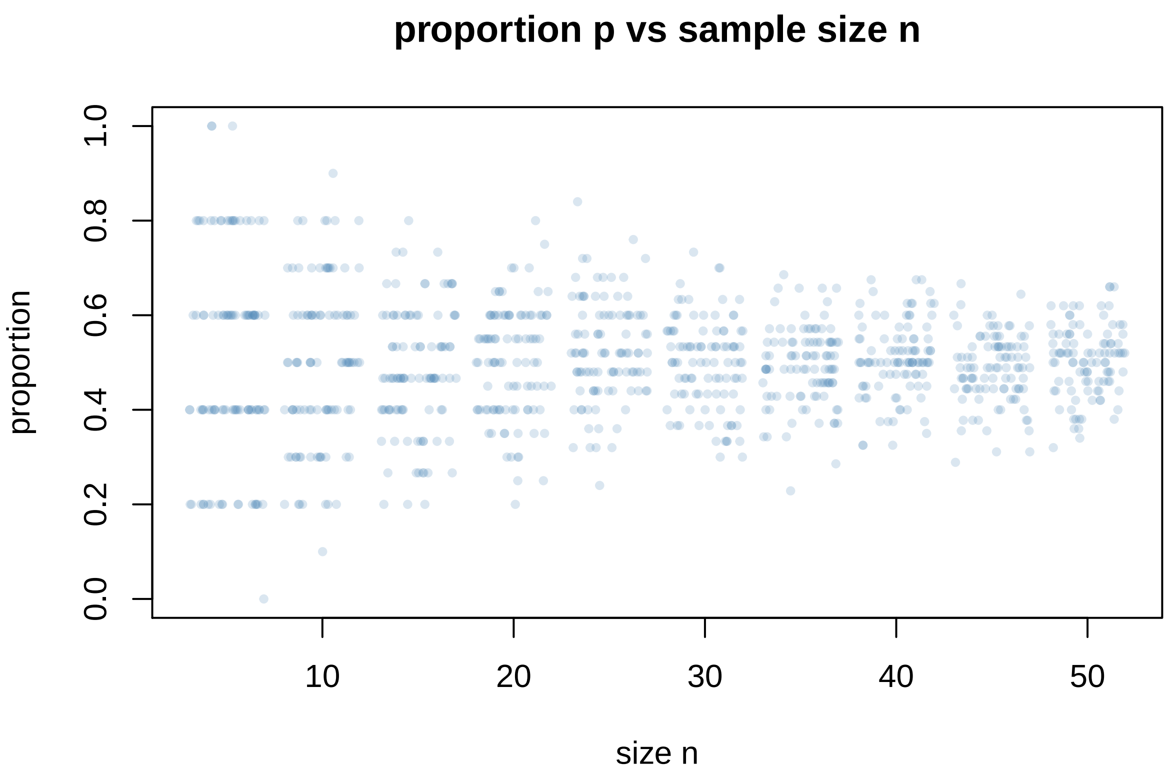
   iii. *(2 marks)* Repeat the production of the above plot, complete with horizontal line, but instead of using `n` as the x variable in the plot, use `jitter(n, 2)`.

      Show your code and resulting plot.

```
Comment on the effect of `jitter()`
```

```r
n_list = seq(5,50,5)
vals<- sapply(n_list, function(n) rbinom(100,n,0.5)/n)
vals1 = unlist(vals)
n1 = rep(seq(5,50,5),each = 100)
n2 = jitter(n_for_plot,2)

plot(jitter(n1,2), vals1, ylim = c(0,1), pch = 19, col =adjustcolor("steelblue", 0.2), cex = 0.5,
     xlab = "size n",ylab = "proportion",main="proportion p vs sample size n")
```



- The effect of jitter add random variation to the values of n.
- Then there can be an estimations of the proportions for sample sizes other than [5,10,...,50]

iv. *(1 mark)* Based on the either of the above plots, what do you conclude about the distribution of binomial proportions as n increases?

- As n increases, the distribution of binominal propertions moves towards the real probability (which is 0.5 in this case).

g. *(6 marks)* With supporting reference to any/all suitable discoveries you have made in the above questions, comment on each of the following:

i. Which values for the true binomial probability p are hardest/easiest to estimate from a sample. Why?

- P = 0.5 is hardest to estimate because from the plot, P = 0.5 gives the largest standard deviation when keeping sample size n constant.
- P = 0.1 is hardest to estimate because from the plot, P = 0.1 gives the smallest standard deviation when keeping sample size n constant.

ii. Law of large numbers? What can you say about the effect of increasing sample size n on the quality and/or interpretation of your estimate $\hat{p}$ of p?

- The average of the results obtained from a large number of trials should be close to the expected value.(Wikipedia, https://en.wikipedia.org/wiki/Law_of_large_numbers)
- Therefore, increasing the sample size n will make a more precise prediction of $\hat{p}$ (closer to p)

iii. Law of small numbers? What can you say about the effect of decreasing sample size n on the quality and/or interpretation of your estimate $\hat{p}$ of p?

```
* Small number of trials can give impercise result(far away from the expectation).
* Becaus there are chances that not all possibility are covered with a shamm sample size,
* likelihood of a Type II error skewing the results can be increased.
```